

LISTING OF CLAIMS

Claim 1. (Currently Amended) A method for operating three-dimensional (3D) application software to generate a 3D stereoscopic vision display, wherein the 3D application software is of the type that provides a 3D output signal to a display driver for a 3D graphics rendering device to generate a display output appearing to be three-dimensional for display on a two-dimensional (2D) screen display, comprising:

(a) running the application software in its normal mode to generate the 3D output signal which is normally sent from the application software to a display driver for a 3D graphics rendering device for generating a 3D display output on a 2D screen display;

(b) intercepting the 3D output signal from the application software and redirecting said 3D output signal to a pseudo driver, wherein said pseudo driver generates from said output signal a left image view signal and a right image view signal that is stereoscopically offset from the left image view signal; and

(c) said pseudo driver providing said left image view signal to a first 3D graphics rendering device and said right image view signal to a second 3D graphics rendering device, and using said first and second 3D graphics rendering devices for separately rendering in tandem left and right image views for display in a 3D stereoscopic vision display device;

wherein the pseudo driver generates a 3D stereoscopic vision display using one physical graphics card with dual graphics generator card heads for separately rendering right and left image view for the 3D stereoscopic vision display.

Claim 2. (Previously presented) A method according to claim 1, wherein the 3D stereoscopic vision display device is selected from the group consisting of head-mounted "stereo vision" goggles, head-mounted 3D display device, and a stereo vision monitor.

Claim 3. (Original) A method according to claim 1, wherein the 3D application software is a 3D video game software which provides 3D game data output.

Claim 4. (Previously presented) A method according to claim 3, wherein the intercepting

and redirecting of the 3D game data is obtained by providing a wrapper for the game software's native API display driver and replacing with stereoscopic pseudo driver display function calls linked under the same name as the game software's native API display driver for a 2D display.

Claim 5. (Original) A method according to claim 4, wherein the wrapper supports a selected one of the following group of native API formats: Glide; OpenGL; and DirectX.

Claim 6. (Canceled)

Claim 7. (Canceled)

Claim 8. (Original) A method according to claim 3, wherein the intercepted 3D game data is stored in a 3D data recorder for later play back.

Claim 9. (Original) A method according to claim 8, wherein the recorded 3D game data are transmitted or downloaded through an online interface to a remote user.

Claim 10. (Original) A method according to claim 3, wherein the intercepted 3D game data is combined with other 3D content using a mixer and a dual rendering system.

Claim 11. (Original) A method according to claim 10, wherein the dual rendering system is kept running while switching between different game software.

Claim 12. (Original) A method according to claim 3, wherein another pseudo driver operates on the 3D game data in tandem with the pseudo 3D display driver.

Claim 13. (Original) A method according to claim 12, wherein the other pseudo driver is a stereo sound or a directional force feedback driver.

Claim 14. (Original) A method according to claim 12, wherein the video game software is run with one or more tracking devices for input from the player.

Claim 15. (Currently Amended) A method of generating a 3D multi-view display operable with three-dimensional (3D) application software of the type that provides a 3D output signal from the application software to a display driver for a 3D graphics rendering device to generate a display output appearing to be three-dimensional on a two-dimensional (2D) screen display and which makes display function calls to a native API display driver to provide the 3D display output to a 2D screen display, comprising:

(a) running the application software on a computer in its normal mode to generate a 3D output signal intended for a 2D screen display;

(b) providing a pseudo 3D display driver that links to the native API display driver by intercepting image display function calls to the native API display driver from the application software and redirecting them through the pseudo 3D display driver in order to generate multiple, separate image views and provide the image views to respective ones of a corresponding multiple 3D graphics rendering devices for a multi-view 3D display;

wherein the pseudo 3D display driver generates right and left eye image views using separate graphics generator cards for rendering the right and left image views, and provides the right and left eye image views to respective right and left graphics rendering devices in parallel for converting the right and left eye views into right and left image display outputs, respectively, which are used for a 3D stereoscopic vision display.

Claim 16. (Previously presented) A 3D multi-view display method according to claim 15, wherein the 3D multi-view display is selected from the group consisting of head-mounted "stereo vision" goggles, head-mounted 3D display device, and a stereo vision monitor.

Claim 17. (Previously presented) A 3D multi-view display method according to claim 15, wherein the 3D application software is a 3D video game software which provides 3D game data output.

Claim 18. (Previously presented) A 3D multi-view display method according to claim 17, wherein the pseudo 3D display driver supports a selected one of the following group of native API formats: Glide; OpenGL; and DirectX.

Claim 19. (Canceled)

Claim 20. (Canceled)